

WeGetletters

by Michael W Lucas



**Oh Generous, Grandiloquent, Gratuitous
FreeBSD Journal Letters-Answerer:**

FreeBSD 13 comes out any time now. It has a whole bunch of features I'm eager to get my hands on, but I'm leery of a brand-new release. Any advice on when I should upgrade?

And do they really pay you in gelato?

Thanks,

—Doesn't Need Outages

DNO,

I know this story.

Early in your career you fell under the tutelage of a grizzled sysadmin, the sort who lost an eye in the Unix Wars, detonated a lobe of his liver bootstrapping the K&R C compiler, and kept a copy of the Alpha boot loader encoded in the knots of his flowing gray beard. You asked him this same question about some other new release. He picked up the *Free the Berkeley 4.4* coffee mug where he kept the knucklebones of the last Ultrix salesman who dared radiate body heat in his artisanally cooled datacenter, gave it a good shake, and cast the bones across his desk to read the wisdom therein. Between the roar of the racked servers all around and the way he'd wrecked his vocal cords screaming at the University of Minnesota's Gopher developers over changing their server to a paid license, you had to listen carefully to sieve his hoarsely whispered wisdom from the noise.

"Never, *never* install a .0 release."

That's the sort of thing that makes quite an impression on a young sysadmin.

In Old One-Eye's defense, that was unsurpassed wisdom in the Dialup Age. The fastest way to download an operating system release was to get a backup tape by mail order. Critical patches were distributed by Usenet, if you were lucky enough to have an account on a site with a mighty 1.544-megabit uplink. Inexpensive servers cost several thousand dollars, or a transplant-ready liver if you could find an insufficiently cautious salesman who hadn't already wrecked theirs.

It's a different world now. For one thing, we have salespeople. And they've all been warned about the liver thing.

If you're eyeing a .0 release today—you're already too late.

Modern operating systems are public, exactly like a sleazy Hollywood star's collection of intimate infections. The time to find problems is before the release. I don't care what flavor of Unix you run, they're public. Even closed-source Unix developers give their customers access to pre-release media, though I'm certain I don't know why you'd want to grant them more help than

your outrageous license fees. The developers have asked, begged, cajoled, pleaded, and threatened their user base to test release candidates, in-progress versions, snapshots, and patches for months or years. And here you are, asking if you can trust the finished product?

You selfish dweeb.

Grab the most recent snapshot, release candidate, or whatever's the latest and greatest, and try it in your environment. Configure it with all the debugging and prepare for kernel dumps. Test your applications under load. Tell the developers what worked and what doesn't.

If you're reading this right after 13.0 came out and are all sorts of relieved that you don't have to do this work, guess what? A pre-pre-release 14.0 is available this very moment! Or maybe all the good topics for PhD theses in Irrelevant American Authors have been taken, but desperation has driven you to delve into the moribund, unrecognizable text archives of a long-telepathic Journal to identify the moment when my descent into ferality crossed into forthright malignance, and release $Ax67.\pi r^2$ is now in development. Whatever the case, there's a forthcoming release available for testing.

No, I'm not saying that you should deploy the release candidates and development versions on every host across your environment. People should, but anyone asking this question shouldn't. You're not equipped.

Testing development releases requires not only sysadmin skills, but sysadmin practices. What's the difference? *Skill* means you know how to do the things you should do. Practice means you perform those things. You need backups. You need to know that those backups can be restored. You must not only know how to submit bug reports, you need to be comfortable submitting them. The whole point of running one of these early versions is to report on bugs.

For your own sanity, you need to deploy development versions intelligently.

Don't slam development releases onto every web server in the cluster. Pick one or two. Put them in the load balancing pool. See what happens. Compare responsiveness under similar loads. Configure them to automatically dump and reboot in case they panic. If they're running ZFS, keep known, good boot environments on hand. While "the kernel panics every one thousand sixty-three seconds, here's the text dump" is eminently valuable, there's no need to live with that once you've identified the problem.

If you keep working it, you'll eventually learn that you can run development versions everywhere. People do. You can become good enough to join them.

Yes, this requires allocating time and hardware. That's cheap. If you doubt me, go price sufficient Oracle Solaris licenses and servers to host your environment. In a big enough company, you can hire official testers and still save enough to feed the staff pizza and beer every day for lunch. Just be sure you give the Oracle rep a burner phone number and an email address in a burner domain name, because like the Terminator, they will never stop, never show mercy or pity, and never tire until they own your scrawny, underfed soul.

You know what improves your soul? What makes your soul blossom? What develops greatness of character and mind? What sharpens your sysadmin chops until none can stand against you?

Testing development versions of the software you depend on.

Deploying them. Using them day-to-day. Providing feedback for the developers. Bug reports are great, but so is "I'm running the latest in production to serve four and a half trillion HamsterSoft queries a second, and it's going great." Negative results are still results.

You know what else improves your soul?

Paying your Letters Column guy his gelato. I had expected George to settle up at BSDCan in spring 2020, but he never showed up. Maybe he'll be there in 2021. If he doesn't come through, though, I'll have to write it off as a bad debt.

These columns might get a little cranky if that happens. Consider yourself warned.

Have a question for Michael?
Send it to letters@freebsdjournal.org



MICHAEL W LUCAS's most recent books include *SNMP Mastery*, *Cash Flow for Creators*, and *Drinking Heavy Water*, plus a bunch more at <https://mwl.io>. Under no circumstances is he allowed near users.



The FreeBSD Project is looking for

- Programmers • Testers
- Researchers • Tech writers
- Anyone who wants to get involved

Find out more by

Checking out our website

freebsd.org/projects/newbies.html

Downloading the Software

freebsd.org/where.html

We're a welcoming community looking for people like you to help continue developing this robust operating system. Join us!

Already involved?

Don't forget to check out the latest grant opportunities at freebsd.foundation.org

Help Create the Future. Join the FreeBSD Project!

FreeBSD is internationally recognized as an innovative leader in providing a high-performance, secure, and stable operating system.

Not only is FreeBSD easy to install, but it runs a huge number of applications, offers powerful solutions, and cutting edge features. The best part? It's FREE of charge and comes with full source code.

Did you know that working with a mature, open source project is an excellent way to gain new skills, network with other professionals, and differentiate yourself in a competitive job market? Don't miss this opportunity to work with a diverse and committed community bringing about a better world powered by FreeBSD.

The FreeBSD Community is proudly supported by

